Amendments to the Specification

Please add the following new paragraphs at the end of page 3:

Figure 5 illustrates one embodiment of a network.

Figure 6 illustrates one embodiment of a super frame data structure to send data over backplane links.

Figure 7 illustrates one embodiment of the overhead portion of a super frame including control messages.

C1

Figure 8 illustrates an example of a super cell structure.

Figure 9 illustrates one embodiment of super cell packing.

Figure 10 illustrates early termination of the received clock domain.

Figure 11 illustrates two SONET chips connected via their telecommunications buses through the backplane.

*Please replace paragraph **[0012]** with the following amended paragraph:*

[0012] For example, physical layer framer 110 can convert data between Synchronous Optical Network (SONET) frames and an internal cell format that is used by the components of the card of Figure 1. The SONET standard is described in the American National Standards Institute (ANSI) standards T1.105 and T1.106 and in the Bellcore Technical Recommendations TR-TSY-000253. Other conversions can also be supported. Examples of conversions to an internal cell format are described in greater detail below.

C2

*Please add the following new paragraphs after paragraph **[0046]**:*

[0046.1] A network box or system, that implements the functionality of one or more of a switch, ADM, crossconnect (e.g., TDM) is described. In one embodiment, the network box utilizes a full mesh backplane that provides a serial link interconnect between each line card in the system with every other line card in

C3

the system. **Figure 5** illustrates one embodiment of a network box. Referring to Figure 5, line cards 501_1-501_N are shown coupled to line cards 502_1-502_N via backplane 510. Backplane 510 comprises a full mesh interconnect in which each of line cards 501-501_N has a dedicated connection to each of line cards 502-502_N. Note that line cards 501_1-501_N and line cards 502_1-502_N are shown arranged with respect to both sides of backplane 510; however, such cards are typically positioned in a cabinet side by side connected to backplane 510 along the same edge of each card.

[0046.2] A backplane protocol is used by transceivers on the line cards to transport data and control information between each other over the full mesh interconnect. The backplane protocol described herein accommodates both TDM and block, or packet, data traffic types so that the fully meshed interconnect operates as a packet and TDM switch fabric using the same set of high speed links.

[0046.3] In one embodiment, the protocol described provides a mechanism whereby control channels between elements within a switch can be implemented in an integrated manner within the same link used for data. The presence of these control channels combined with the fully meshed interconnect allows for the implementation of a distributed switch architecture.

C3

[0046.4] In one embodiment, the protocol allows for variable sized packets on the backplane links. This allows the links to maintain full throughput regardless of the arriving packet sizes and under-utilization if the backplane link will be avoided. In one embodiment, the protocol also allows the backplane links to be clocked independently from any of the timing references used on the interfaces out of the network box. This is accomplished by transferring data on the link that is marked as " don' t care". This data is referred to herein as stuffing. The receiver throws away, or ignores, that data, and thus, the receiver in avoiding processing that data can use that time to accommodate for differences in the timing references of the transmitter on one card and the receiver on the other.

[0046.5] In one embodiment, the protocol described herein allows for integrating incremented protocol upgrades. New line cards may utilize new versions of the backplane protocol. These new cards may be designed to support older versions of the protocol as well. In one embodiment, backplane links to/from older cards use an older version of the protocol, while links

to/from new cards use the new version. In this fashion, new types of cards can be introduced into network boxes without having to remove older cards.

[0046.6] **Figure 6** illustrates one embodiment of a frame data structure to send data over backplane links. Referring to Figure 6, the data structure comprises a frame overhead 601, packet/TDM data 602, and stuffing 604. In one embodiment, each of frame overhead 601 and packet/TDM data 602 of the frame comprises a fixed number of 16-bit words. A line card splits the data transported into 16-bit words and collects a number of such words into the frame structure.

[0046.7] Stuffing 604 follows the frame to accommodate for frequency differences between the system frame pulse and the backplane reference clock as well as for frequency differences between the backplane reference clocks of two cards if not the same.

[0046.8] In one embodiment, the frame is sent over the backplane links as a 125us structure with stuffing 604 at the end. The super frame and stuffing 604 have a duration of a system frame pulse. In one embodiment, transmission of the packet started with a system wide 8kHz SONET compliant pulse 610. That is, frame pulse 610 is derived from a SONET compliant clock.

C3

[0046.9] In one embodiment, the super frame overhead has the following functions: framing synchronization; bit/byte/word synchronization; checksum for link performance monitoring; provide packet pointer (start of new packet); data channels for card-to-card communication. In another embodiment, the super frame overhead also functions to distribute TDM/cell allocation information.

[0046.10] **Figure 7** illustrates one embodiment of the overhead of the frame of Figure 6. Referring to Figure 7, frame 700 includes framing pattern slot 701, version/coding slot 702, checksum slot 703, control slot 704, block data pointer slot 704, message count slot 706, and a reserve/undefined slot 707, followed by a number of slots for control messages 710 and a number of slots 711 that are reserved or unused.

[0046.11] The information in framing pattern slot 701 is used by the receiver on the line card to locate the start of the frame and to align the bytes and/or words. In one embodiment, the frame is started a fixed delay after a system pulse (e.g., 8KHz). Therefore, the receiver knows approximately when to look for the pattern.

Version/coding slot 702 contains version control information. In one embodiment, version control information enables changes in the frame structure with respect to backward compatibility. For example, newer versions always support older formats. Once the version information is received by a receiver, the receiver may use the proper coding or scrambling that is associated with that version.

[0046.12] Checksum slot 703 contains the check sum that is used for performance monitoring of the link.

[0046.13] Control slot 704 contains control related information. In one embodiment, control slot 704 provides locations for TDM/block data allocation bits that allow for performing synchronization procedures when changing the allocation between TDM and packet data on a backplane link. In one embodiment, the new allocation is filled in by both egress and ingress cards before writing an update bit on the ingress card. When the update bit is written on the ingress card, the next frame uses the new allocation and a synchronization message is sent.

[0046.14] Block data pointer slot 705 contains a pointer to the start of a new block data in the frame. This pointer is included because it can not be assumed that the last block data in the last super frame was transmitted in full. By having the pointer, the start of a first new block data in each frame can be located.

C3

[0046.15] Message count slot 706 contains information indicative of the number of control messages that are valid in the current super frame.

[0046.16] The reserve/undefined slots 707 are currently designated for future use; however, in another embodiment they may be used for a variety of functions. The same is true of the unused/reserved slots 711.

[0046.17] Control message slots 710 provide transport for low latency control channels for controls, such as, but not limited to, flow control, protection switching control data, etc.

[0046.18] The second portion of the frame is for transporting the packets and TDM data. In one embodiment, the packet/TDM portion consists of a number of channels, each carrying a STS-1 rate signal (approximately 52 Mbits/s). The number of channels depends on the speed used for the backplane link (i.e., link speed). For instance, a 3.125 Gbits/s link speed gives approximately 60 channels, or slots. For 60 channels, each of the channels can be allocated to either TDM data, packet data, or control data. In one

embodiment, there are 6 channels dedicated to packet data, 6 channels dedicated to packet control data and 48 channels dedicated to packet and/or TDM data in each frame. Other allocations are possible, including those due to having less than 60 channels.

[0046.19] In one embodiment, to keep the latency low, the channels are interleaved on a 16-bit level, with 16-bits from each channel forming a "s uper cell". **Figure 8** illustrates an example of a super cell structure. Referring to Figure 8, supercell 800 comprises 48 slots for TDM/packet data, 6 slots dedicated to packet data, and 6 slots dedicated for overhead. In one embodiment, each slot not allocated to TDM or overhead is allocated to packet data. In one embodiment, there are 805 supercells in each frame structure, or in one cycle. **Figure 9** illustrates one embodiment of super cell packing. Referring to Figure 9, frame overhead 901 is followed by supercells 0-404, which is followed by stuffing 902. The supercells, supercell(0)–supercell (404), are placed one after another to fill a frame. In one embodiment, the super cells are put one after another until 810 bytes are put in each channel (to match the STS-1 rate).

C3

[0046.20] Although there are 405 super cells, the channels may change to compensate for changes in the link speed over the interconnect. In other words, the number of channels may change while the number of super cells remains the same. Thus, for any one channel the latency and throughput stay the same regardless of the number of channels.

[0046.21] The stuffing in the end of the super frame structure allows for adjusting the super frame rate to match the TDM data rate, i.e., 125 us period, over long periods. The stuffing words also make it possible to terminate the received clock domain very quickly, which is critical for an FPGA implementation. The stuffing accounts for slight variations in the clocks between the transmit and receive domains. This is because the stuffing is not received for processing. Therefore, if the processing rate on the receive card is slower than the data is being sent, the fact that the stuffing is not processed allows time for a slower receive card to process the data without incurring errors due to the small amount of difference in the clock speeds in the transmit and receive domains. In essence, this enables the format to be independent of the clock.

[0046.22] In one embodiment, the first word of the overhead is selected so that a single bit error in the stuffing does not result in

the two being the same. Therefore, if an error occurs in the stuffing, a line card will not confuse the stuffing with the start of a super frame.

[0046.23] **Figure 10** is a block diagram of one embodiment of an interface on a line card to send and receive information. Referring to Figure 10, receiver 1005 receives frames in the form of a bit stream from another line card via a link on the backplane. The data is clocked-in using a receiver (Rx) clock 1021. The clocked-in data is forwarded to frame pattern matching block 1004 that performs frame pattern matching and word alignment on the received bit stream in a manner well-known in the art. After frame pattern matching and word alignment, descrambler 1003 performs descrambling in a manner well-known in the art, and stores the descrambled data into FIF0 1002 using Rx clock 1021 as a write clock. In one embodiment, descrambler 1003 performs 2-stage synchronous descrambling, including performing scrambling according to SONET scrambling $1+x^6+x^7$ and performing the scrambling according to the following equation: $(1+x^{43})$. No stuffing words are written into FIF0 1002. Demapper 1001 reads data from FIF0 1002 according to a read clock and performs a demapping (e.g., sorting) operation to produce a cell data stream 1031, an overhead data stream 1032, and a TDM data stream 1033. In one embodiment, the read clock comprises the transmit (Tx) clock 1020 used for sending frames and is the clock for demapper 1001.

Q3

[0046.24] For transmission, mapper 1011 receives a cell data stream 1041, an overhead stream 1042 and a TDM data stream 1043 and combines them into a single data stream. Scrambler 1012 receives the stream of frames and scrambles them. In one embodiment, scrambler 1012 performs a 2-stage frame synchronous scrambling. The scrambled frames are sent and transmitted by transmitter 1013. Each of frame mapper 1011, scrambler 1012, and transmitter 1013 are coupled to receive, and operate based upon, at least in part, Tx clock 1020.

[0046.25] In one embodiment, the stuffing is done at a 16-bit word level resulting in jitter in the TDM data. However, this jitter will be removed after a buffer (coupled to the TDM output of a demapper) that takes the data into the " telecom" clock domain. By stuffing with 16-bits, the bit/byte/word alignment does not have to be redone after it is found.

[0046.26] **Figure 11** illustrates two SONET chips connected via their telecom buses through a backplane. For simplicity, data is shown only going in one direction. Therefore, reference to ingress and egress given in the following example are not indicative of the sole function of a device and may be switched when the data direction is revised.

[0046.27] Referring to Figure 11, both backplane ASICs are coupled to receive the system frame pulse. This pulse is used both for super frame synchronization on the backplane and for the frame pulse indications to the SONET chips. As shown in Figure 11, the first pulse to be generated (from the system frame pulse) is the ingress SONET frame pulse. This pulse causes SONET chip 1101 to output the start of the SONET frame (first byte after J0) on the drop telecom bus 1111. This data is put into small FIF0s (not shown) inside ASIC 1102. At the start of the backplane framing pulse, the super frame is sent out and, at the first TDM slot, data is read out from the TDM ingress FIF0s.

C3

[0046.28] On the egress side, the backplane ASIC 1103 receives the start of the super frame and soon thereafter obtains TDM data. This data is again put into small FIF0s in ASIC 1103.

[0046.29] Some time after the backplane frame pulse, backplane ASIC 1103 generates an egress frame pulse to egress SONET chip 1104. At this time the TDM data is available in the egress FIF0s and can be placed on the add telecom bus 1112.

[0046.30] Ingress SONET chip 1101 adjusts and outputs the SPE pointers (as defined in SONET standard) according to the frame pulse. Egress SONET chip 1104 only needs the frame pulse marker and then performs " normal" SONET pointer processing.

The new paragraphs are from U.S. Patent application number 09/745,982, filed December 12, 2000 and entitled "A BACKPLANE PROTOCOL," which was incorporated by reference in the present application. The figure numbers and reference numerals have been changed to be consistent with the present application. Therefore, no new matter has been added.